Sistemas de Operação / Fundamentos de Sistemas Operativos

Course Overview

Artur Pereira <artur@ua.pt>

DETI / Universidade de Aveiro

Sumário

Sumário

Objectives and outcomes

- Objectives

- Acquired competencies

Objectives and outcomes

- Objectives
  - To present the most important concepts about the internal organization of present day operating systems

- Acquired competencies
  - To gain a good understanding of how multiprogramming works and of the general organization of present day operating systems

Objectives and outcomes

- Objectives
  - To present the most important concepts about the internal organization of present day operating systems
  - To introduce concurrent programming and the most important mechanisms for interprocess communication and synchronization

- Acquired competencies
  - To gain a good understanding of how multiprogramming works and of the general organization of present day operating systems
  - To develop skills for the project and implementation of simple concurrent applications

Objectives and outcomes

- Objectives
  - To present the most important concepts about the internal organization of present day operating systems
  - To introduce concurrent programming and the most important mechanisms for interprocess communication and synchronization
  - To acquaint the students with the Unix internal organization

- Acquired competencies
  - To gain a good understanding of how multiprogramming works and of the general organization of present day operating systems
  - To develop skills for the project and implementation of simple concurrent applications

Objectives and outcomes

- Objectives
  - To present the most important concepts about the internal organization of present day operating systems
  - To introduce concurrent programming and the most important mechanisms for interprocess communication and synchronization
  - To acquaint the students with the Unix internal organization

- Acquired competencies
  - To gain a good understanding of how multiprogramming works and of the general organization of present day operating systems
  - To develop skills for the project and implementation of simple concurrent applications
  - To be able to carry out productive work as a member of a team that develops system programming software

Sumário

Prerequisites

- At the computer architecture level:

- At the programming level:

- At the data structures level:

Prerequisites

- At the computer architecture level:
  - basic notions on computer architecture

- At the programming level:

- At the data structures level:

Prerequisites

- At the computer architecture level:
  - basic notions on computer architecture
  - basic notions on communication protocols with input-output devices (pooled I/O, interrupt driven I/O and DMA based I/O)

- At the programming level:

- At the data structures level:

Prerequisites

- At the computer architecture level:
  - basic notions on computer architecture
  - basic notions on communication protocols with input-output devices (pooled I/O, interrupt driven I/O and DMA based I/O)

- At the programming level:
  - programming skills in C/C++ language at a fair to good level

- At the data structures level:

Prerequisites

- At the computer architecture level:
  - basic notions on computer architecture
  - basic notions on communication protocols with input-output devices (pooled I/O, interrupt driven I/O and DMA based I/O)

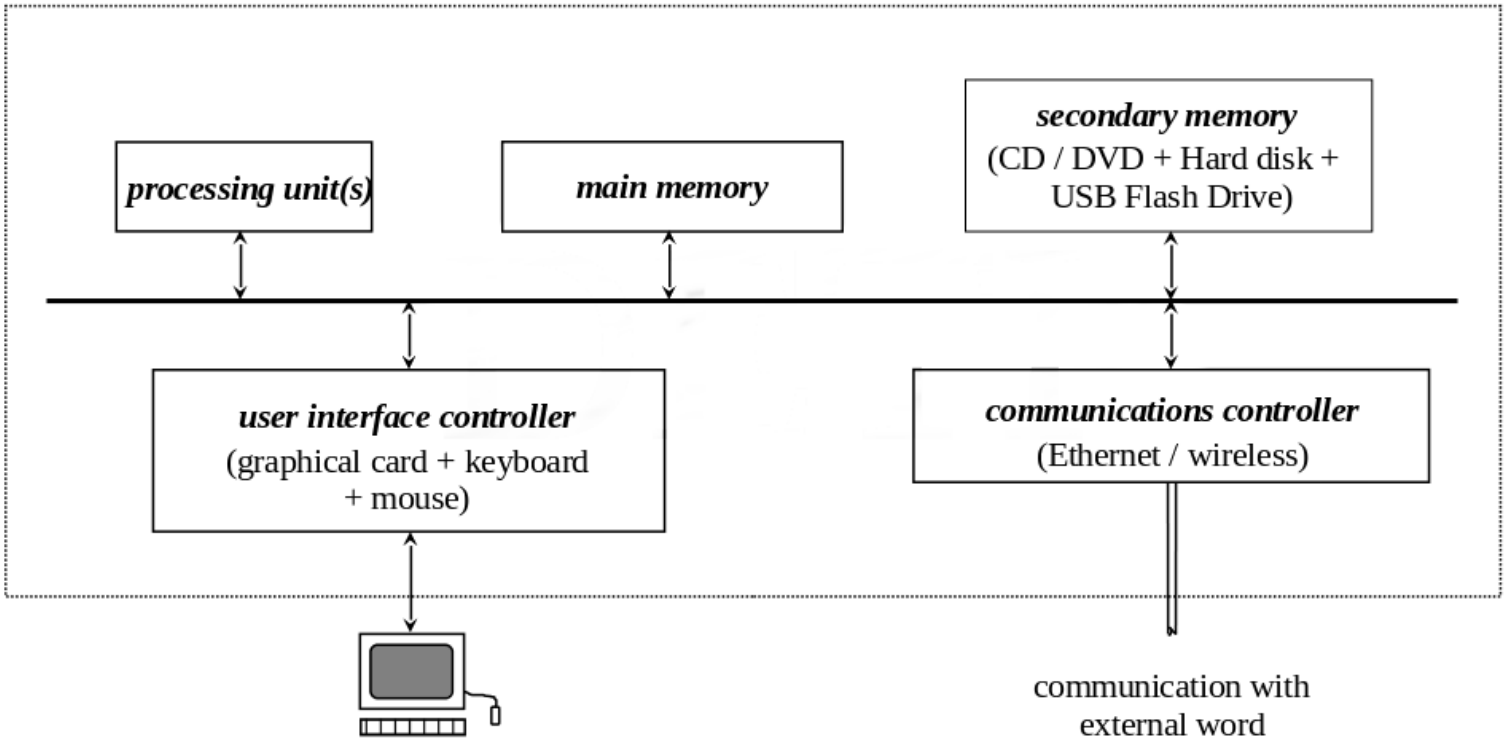
- At the programming level:
  - programming skills in C/C++ language at a fair to good level

- At the data structures level:
  - operational and conceptual knowledge of the most common static and dynamic data structures used to build different types of memory (RAMs, stacks, FIFOs and associative memories)

Sumário

Course contents
Computational system

- Simple view of a computational system:



Course contents
Summary

- Theoretical topics:

- Practical and Lab topics:

Course contents
Summary

- Theoretical topics:
  - Introductory concepts
  - Processor management in multiprogramming
  - Interprocess communication and synchronization
  - Memory management
  - Input / Output
  - File systems
  - Protection and Security (some introductory notions, if possible)

- Practical and Lab topics:

Course contents
Summary

- Theoretical topics:
  - Introductory concepts
  - Processor management in multiprogramming
  - Interprocess communication and synchronization
  - Memory management
  - Input / Output
  - File systems
  - Protection and Security (some introductory notions, if possible)

- Practical and Lab topics:
  - File system development

Course contents
Summary

- Theoretical topics:
  - Introductory concepts
  - Processor management in multiprogramming
  - Interprocess communication and synchronization
  - Memory management
  - Input / Output
  - File systems
  - Protection and Security (some introductory notions, if possible)

- Practical and Lab topics:
  - File system development
  - Concurrent programming, involving inter-process/thread communication and synchronization

Sumário

Bibliography

- Support bibliography:
  - Operating Systems: Internals and Design Principles,
    W. Stallings,
    Prentice-Hall International Editions, 7th Ed, 2012
  - Operating Systems Concepts,
    A. Silberschatz, P. Galvin and G. Gagne,
    John Wiley & Sons, 9th Ed, 2013
  - Modern Operating Systems,
    A. Tanenbaum and H. Bos,
    Pearson Education Limited, 4th Ed, 2015
  - Sistemas Operativos,
    J. Marques, C. Ribeiro, L. Veiga, P. Ferreira and R. Rodrigues,
    FCA, 2012
  - *Lecture Slides*

- The lecture slides are not enough for a robust understanding of the course topics!

Sumário

Practical classes
Schedule

- General schedule:
  - Bash scripting - 1 session
  - File system project - 6/7 sessions
  - Inter-process communication and synchronization (IPC) - 6/7 sessions

Practical classes
Schedule

- General schedule:
  - Bash scripting - 1 session
  - File system project - 6/7 sessions
  - Inter-process communication and synchronization (IPC) - 6/7 sessions

- File system project:
  - Implementation of a file system, including its integration into the Linux operating system

Practical classes
Schedule

- General schedule:
  - Bash scripting - 1 session
  - File system project - 6/7 sessions
  - Inter-process communication and synchronization (IPC) - 6/7 sessions

- File system project:
  - Implementation of a file system, including its integration into the Linux operating system

- IPC and concurrent programming:
  - Exercise on concurrent programming, based on processes, shared memory and semaphores

Practical classes
Schedule

- General schedule:
  - Bash scripting - 1 session
  - File system project - 6/7 sessions
  - Inter-process communication and synchronization (IPC) - 6/7 sessions

- File system project:
  - Implementation of a file system, including its integration into the Linux operating system

- IPC and concurrent programming:
  - Exercise on concurrent programming, based on processes, shared memory and semaphores
  - Exercise on concurrent programming, based on threads, mutexes and condition variables

Practical classes
Schedule

- General schedule:
  - Bash scripting - 1 session
  - File system project - 6/7 sessions
  - Inter-process communication and synchronization (IPC) - 6/7 sessions

- File system project:
  - Implementation of a file system, including its integration into the Linux operating system

- IPC and concurrent programming:
  - Exercise on concurrent programming, based on processes, shared memory and semaphores
  - Exercise on concurrent programming, based on threads, mutexes and condition variables
  - Training exercise for the practical exam

Sumário

Assessment
General rules

- 2 components:
  - theoretical component: 45%, with a minimum of 7.0
  - practical component: 55%, with a minimum of 8.0
- all intermediate marks are rounded to one decimal place

Assessment
General rules

- 2 components:
  - theoretical component: 45%, with a minimum of 7.0
  - practical component: 55%, with a minimum of 8.0
- all intermediate marks are rounded to one decimal place

- Theoretical component with 1 element:
  - written exam, at the exam periods

Assessment
General rules

- 2 components:
  - theoretical component: 45%, with a minimum of 7.0
  - practical component: 55%, with a minimum of 8.0
- all intermediate marks are rounded to one decimal place

- Theoretical component with 1 element:
  - written exam, at the exam periods

- Practical component with 3 elements:
  - a file system project: 20%
  - midterm quiz (on the file system project): 15%
  - practical exam on concurrent programming: 20%
  - Marks above 17 may required some extra work

Assessment
General rules

- 2 components:
  - theoretical component: 45%, with a minimum of 7.0
  - practical component: 55%, with a minimum of 8.0
- all intermediate marks are rounded to one decimal place

- Theoretical component with 1 element:
  - written exam, at the exam periods

- Practical component with 3 elements:
  - a file system project: 20%
  - midterm quiz (on the file system project): 15%
  - practical exam on concurrent programming: 20%
  - Marks above 17 may required some extra work

- Repeating students:
  - Can inherit, but ...

Assessment
Appeal and special exam periods

- In the appeal and special exam periods, the assessment elements are exactly the same
- The following inheritance rules apply:
  - the grade of theoretical exam can be inherited from a previous exam period
  - the grade of practical exam can be inherited from a previous exam period
  - the grades of the file system project and of the midterm quiz can be inherited as a whole from a previous exam period
  - Repeating the file system project and of the midterm quiz involves a new file system, not the sofs20

Assessment
Inheritance rules for repeating students

- By default:
  - grades obtained in previous years are not inherited directly

- However, a grade can be inherited based on the following rules:
  - written exame: 100% of the grade
  - file system project: 90% of the grade, with maximum of 13.5
  - midterm quiz: 100% of the grade
  - concurrent programming exam: 100% of the grade
  - concurrent programming exam from concurrent programming project: 90% of the grade, with maximum of 14.0